

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 05-324396

(43)Date of publication of application : 07.12.1993

(51)Int.Cl.

G06F 11/28

(21)Application number : 04-126417

(71)Applicant : FUJITSU LTD
NIPPON TELEGR & TELEPH CORP
<NTT>

(22)Date of filing : 19.05.1992

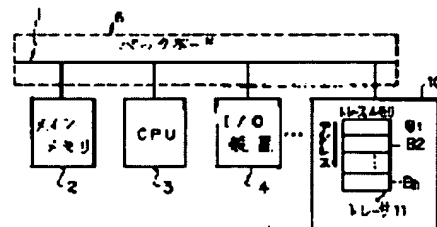
(72)Inventor : NAMITO YUTAKA
KURAYA ETSUJI

(54) PROGRAM TRAVELING HISTORY RECORDING SYSTEM

(57)Abstract:

PURPOSE: To enable the investigation of the combined cause of faults even when the faults occur continuously several times by performing traces several times and storing the execution conditions of a program every fault occurrence relating to the program traveling history recording system for tracing the execution conditions of the program as a hardware.

CONSTITUTION: The storage capacity of a trace memory 10 is divided into plural banks B1, B2,... Bn in the address direction. When the occurrence of the fault is detected while writing data appeared on a processor bus 1 into the prescribed bank, the remaining storage area inside the bank during the write is jumped, the write into the next bank is performed at each fault occurrence and after the write into the all banks is completed, writing data is completed.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's]

(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)特許出願公開番号

特開平5-324396

(43)公開日 平成5年(1993)12月7日

(51)Int.Cl.⁵

G 0 6 F 11/28

識別記号

3 1 0 B 9290-5B

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数5(全9頁)

(21)出願番号 特願平4-126417

(22)出願日 平成4年(1992)5月19日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(71)出願人 000004226

日本電信電話株式会社

東京都千代田区内幸町一丁目1番6号

(72)発明者 波戸 裕

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(72)発明者 倉矢 悦治

東京都千代田区内幸町1丁目1番6号 日

本電信電話株式会社内

(74)代理人 弁理士 大宮 義之 (外1名)

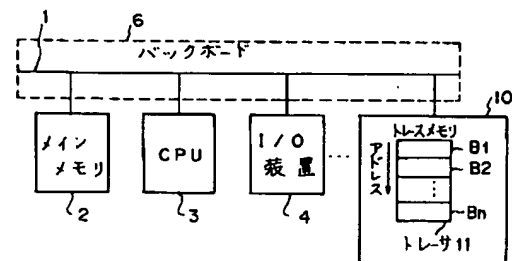
(54)【発明の名称】 プログラム走行履歴記録方式

(57)【要約】

【目的】 本発明はプログラムの実行状況をハードウェア的にトレースするプログラム走行履歴記録方式に関し、トレースを複数回実施して、障害発生毎にプログラムの実行状況を記憶することにより、障害が複数回連続して発生した場合でも、複合的な障害原因の究明を可能とすることを目的としている。

【構成】 トレースメモリ11の記憶容量を、そのアドレス方向に複数のバンクB1、B2、・・・、Bnに分割し、プロセッサバス1上に現れたデータを所定のバンクに書き込み中、障害発生を検出すると、その障害検出毎に、書き込み中のバンク内の残りの記憶エリアを飛び越して次のバンクへの書き込みを行い、すべてのバンクへの書き込みを終了後、データ書き込みを終了する構成とする。

本発明原理を説明するシステム構成図



【特許請求の範囲】

【請求項1】 プロセッサバス(1)上に現れるデータをトレースして、トレースメモリ(11)に記憶し、このトレースメモリに記憶された内容からプログラム実行状況を解析するプログラム走行履歴記録方式において、上記トレースメモリ(11)の記憶容量を、そのアドレス方向に複数のバンク(B₁, B₂, …, B_n)に分割し、上記プロセッサバス(1)上に現れたデータを所定のバンクに書き込み中、障害発生を検出すると、その障害検出毎に、書き込み中のバンクから次のバンクへ変更して書き込みを行い、すべてのバンクへの書き込み終了後、データの書き込みを終了することを特徴とするプログラム走行履歴記録方式。

【請求項2】 上記プロセッサバス(1)上に現れるデータのトレースメモリ(11)への書き込みは、通常時は、トレースメモリ(11)の所定のバンク内にサイクリックに上書きをして書き込みを行い、データがジャンプ命令の時は、トレースメモリ(11)のアドレスカウンタを更新し、この更新したアドレスにデータを書き込むようにしたことを特徴とする請求項1記載のプログラム走行履歴記録方式。

【請求項3】 上記障害検出時にトレースメモリ(11)のアドレスカウンタを更新して、その更新したアドレスに障害発生ポイントを示すデータを書き込むことを特徴とする請求項1記載のプログラム走行履歴記録方式。

【請求項4】 上記障害検出は、トレースメモリ(11)の属するプロセッサで発生する内部障害と、他のプロセッサで発生する外部障害とを検出することを特徴とする請求項1記載のプログラム走行履歴記録方式。

【請求項5】 上記トレースメモリ(11)の内容を外部表示装置(16)で表示し、その表示内容からプログラム走行状況を解析するようにしたことを特徴とする請求項1記載のプログラム走行履歴記録方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、電子交換機やコンピュータなどのプログラムの実行状況を、ハードウェア的にトレースし、そのトレース結果を装置内のメモリに記憶して、故障発生時にメモリに記憶されているプログラムの実行履歴を調べることにより故障原因の究明を行うプログラム実行履歴記録方式に関する。

【0002】

【従来の技術】近年の電子交換機やコンピュータは、大規模LSIの採用によるハードウェアの複雑化あるいは多様なサービスを提供するためのソフトウェアの膨大化により、一旦故障が発生すると、その原因究明に時間がかかることが多く、また故障から復旧までに長時間必要となる傾向にある。

【0003】このため、ハードウェアの面からソフトウ

エアの動きをトレースし、故障発生時に直ちにそのトレース内容を解析して、障害の早期回復を図る必要がある。従来上記したようなハードウェアによってソフトウェア(プログラム)の動きをトレースする方式の実用化は少なく、仮りにあるとしても、従来方式は、障害(Emergency,以下EMAと略称する)によって生ずる一度のトリガでトレースを終了する方式であった。

【0004】図6は従来方式を説明するためのシステム構成図である。同図において、1はプロセッサバスで、このプロセッサバス1には、ある1つのプロセッサとして、メインメモリ2、CPU3、入出力(I/O)装置4などの他にプログラムの実行状況をトレースするプログラム走行履歴記録装置(以下トレーサという)5が接続されている。上記プロセッサバス1はバックボード6(プリント基板)上に設けられており、また、上記メインメモリ2、CPU3、I/O装置4、トレーサ5はそれぞれのパッケージとして構成され、上記バックボード6のプロセッサバス1にコネクタなどで接続されるようになっている。そして、このプロセッサバス1には、バススラバなどを介して他のプロセッサ(図示せず)も接続されている場合もある。

【0005】このような構成において、上記CPU3はメインメモリ2に格納されたプログラムに基づいて所定の処理を行うが、このとき、トレーサ5は、プロセッサバス1上の処理アドレス及びデータをモニタし、それを自己のトレースメモリ(ここで図示せず)に書き込んで行く。そしてEMAを検出するとその時点でトレースを終了する。

【0006】

【発明が解決しようとする課題】このように従来方式では、一度のEMA検出でトレースを終了していた。しかしながら、電子交換機のように、障害の発生した際、予備系に自動的に切り替えて、処理を継続するシステムでは、予備系への切り替えでも引き続きEMAが発生する場合が考えられ、1回のEMA検出でトレースを終了する方式では問題があった。

【0007】本発明は、EMAが複数回連続して発生した場合でも、各々のEMA発生時点でのプログラムの動きを記録し、複合的なEMA原因の究明を短時間にかつ適確に行えるプログラム走行履歴記録方式を目的としている。

【0008】

【課題を解決するための手段】図1は本発明原理を説明する図であり、システムの構成としては、前記従来方式の説明の際に参照した図6とはほぼ同様であるので、同一部分には同一符号を付してある。すなわち図1において、1はプロセッサバスで、このプロセッサバス1にはメインメモリ2、CPU3、I/O装置4などが接続され、さらにこの発明によるトレーサ10が接続されている。

【0009】上記トレサ10にはトレースメモリ11が設けられている。このトレースメモリ11は、その全体の記憶容量がアドレス方向に複数のバンクB₁, B₂, ..., B_nに分割されている。

【0010】

【作用】このような構成において、CPU3がメインメモリ2に格納されたプログラムに基づいて処理を行うと、トレサ10はプロセッサバス1上に出現する上記プログラムのアドレスを自己のトレースメモリ11に書き込んで行く。

【0011】このアドレスの書き込みは、通常時は、トレースメモリ11のバンクB1内にサイクリックに上書きをしながら書き込んでいく。この書き込み中にEMAが発生し、EMAが検出されると、その時点でバンクB1への書き込みを中止し、次のバンクB2への書き込みを開始する。そして上記同様、プロセッサバス1上に現れるアドレスをバンクB2にサイクリックに上書きしながら書き込んで行く。ここでまたEMAが発生すれば、バンクB2への書き込みを中止して次のバンクB3への書き込みを開始する。

【0012】このようにして、所定のバンクへのアドレスの書き込み中にEMAが発生すると、その時点でそのバンクへの書き込みを中止して、残りの記憶エリアをジャンプして次のバンクへの書き込みを開始する。このように、EMAが発生したとき、そのバンクの残りの記憶エリアをジャンプして次のバンクに移るのは、EMAの発生する直前までの書き込み内容を消去せずに残すためである。すなわち、たとえばバンクB1への書き込みを行う場合、バンクB1内の最初のアドレスから順に最終アドレスまで書き込んで行って、再び最初のアドレスから書き込みをした時点で、EMAが発生したとすると、その時点で書き込みを中止したとしても、次の書き込みをバンクB1の続きのアドレスから書き込んでしまうと、EMAが発生する直前までの書き込まれた情報が消去されてしまうからである。これを防止するためにEMAが発生した時点でそのバンクへの書き込みを中止して、書き込まれた情報の履歴を残した状態とするため、ジャンプして次のバンクに書き込みを移している。

【0013】以上のようにして、トレースメモリ11の各バンクB1, B2, ..., B_nへの書き込みが終了したあと、パソコンなどを接続して、このトレースメモリの内容を読み出して、それを解析することにより、プログラムの動きを知ることができ、障害原因などを究明することができる。

【0014】このように本発明では、プログラムの実行状況をトレースする際、障害が発生する毎にその障害発生直前までの履歴を残すことができ、複合的な装置故障の原因究明に有効なデータを得ることができる。

【0015】

【実施例】図2はこの発明を実現するための実施例の構

成図であり、図1と同一部分には同一符号を付してある。図2において、トレサ10は、バスインタフェース部12、EMA検出回路13、外部インタフェース部14、メモリ制御回路15、前記したトレースメモリ11から構成されている。また、16は上記外部インタフェース部14に接続されて、トレースメモリ11の内容を表示するパソコンなどの外部表示装置である。

【0016】プロセッサバス1にはトレースメモリ11の他に、前記したように、プログラムなどが記憶されているメインメモリ2、CPU3、I/O装置4などが接続されており（この図3ではこれらは図示されていない）、CPU3がメインメモリ2に格納されているプログラムを逐一読み出して実行するとき、このプロセッサバス1上には、そのアドレス/データ情報が存在する。また、I/O装置4からのCPU3へのデータやコマンドもプロセッサバス1上に現れる。

【0017】上記バスインタフェース部12は、上記プロセッサバス1を常時モニタし、アドレスを取り込むものであるが、実際には、プロセッサバス1上のアドレスストロブ信号(ADS)を検出することによりこのADSに同期してアドレスを取り込むものである。そして、プロセッサバス1上のアドレスを取り込むと、メモリ制御回路15を起動して、そのアドレスがトレースメモリ11の所定のバンクに書き込まれるようになっていく。

【0018】EMA検出回路13は、自己プロセッサ内で発生したEMA（これを内部EMAという）による内部EMA検出信号と、他のプロセッサで発生したEMA（これを外部EMAという）による外部EMA検出信号とを取り込んで、その論理和をとって出力するものである。この内部EMA検出信号または外部EMA検出信号によりメモリ制御回路15は、トレースメモリ11へのアドレス書き込み動作を制御する。なお、上記外部EMA検出信号は専用のラインLにより、自己プロセッサ以外の他のプロセッサから送られてくる。

【0019】また、トレースメモリ11は図3に示すように、この実施例では、256KW×28bitの容量とし、それをアドレス方向に4つに分割して、64KW×28bitの4つのバンクB₁, B₂, B₃, B₄に分けて使用する。このトレースメモリ11の詳細については後述するが、概略的な動作としては、前記本発明原理で説明したように、通常は、プロセッサバス1上のアドレス情報をバンクB1の最初のアドレスから最終アドレスまでの間でサイクリックに上書きしながら順次書き込んで行き、EMAが発生すると、その時点で書き込みを中止して、残りの記憶エリアをジャンプして次のバンクB2に書き込みを移すという動作を行う。

【0020】このようにして、トレースメモリ11の4つのバンクB1~B4への書き込みが終了すると、パソコンなどの外部表示装置16を接続して外部インタフェ

10

20

30

40

50

ース部14により、トレースメモリ11の内容を読み出してディスプレイ画面上に表示する。これを解析することにより、プログラムの動きを知ることができ、障害原因の究明を行うことができる。

【0021】次に上記した動作を図4のフローチャートを参照しながらさらに説明する。まず、バスインタフェース部12はプロセッサバス1をモニタし、ADS信号を検出(処理S1)し、種々の動作を行うためのタイミングT0、T1、・・・を作る。そして装置の初期設定を行う(処理S2)が、この初期設定はタイミングT0からT1の間に行われる。次に、タイミングT2でアドレス情報を内部に取り込み(処理S3)、同じタイミングでこのアクセスが命令フェッチ(I-F)なのか、ジャンプ成立直後の命令フェッチ(J-I-F)なのか、それ以外なのかの判定を行う(処理S4)。この判定はプロセッサバス1上のプロセッサステータス信号により行う。

【0022】命令フェッチ(I-F)である場合は、トレースメモリ11の所定のバンクにT6のタイミングで上書きしながら順次書き込んで行く(処理S5)が、プログラムの軌跡を追う場合、ジャンプ元アドレスからその軌跡を追うのが一般的である。したがって、ジャンプ成立直後の命令フェッチ(J-I-F)を実行すると、この場合、T4のタイミングでトレースメモリ11のアドレスを指示するアドレスカウンタの値を+1し(処理S6)、その+1をしたアドレスでT6のタイミングでトレースメモリ11に書き込む(処理S7)。たとえば、アドレスカウンタの値が100番地、ジャンプアドレスが200番地、・・・の場合、100番地のアドレスに+1をして101番地とし、200番地のジャンプアドレスをトレースメモリ11に書き込む。そしてこの状態で通常の命令フェッチが出されれば、その命令を200番地、201番地、202番地、・・・というように上書きをして書き込んで行く。

【0023】このように、ジャンプ命令があると、アドレスカウンタを+1してデータ(実行アドレス)を書き込んで行き、ジャンプ命令がないときは同じアドレスにデータを上書きをして行く。したがって、この処理ループを繰り返すことにより、ジャンプ命令のアドレスだけがトレースメモリ11に履歴として残されて行くことになる。

【0024】また、上記I-FかJ-I-Fかその他の判定処理(処理S4)と同じタイミング(T2)にて、アドレス情報の取り込み(処理S3)が行われ、この取り込んだ時EMAコマンドが検出されると(処理S8)、EMAフリップフロップをセットする(処理S9)。このとき、外部EMA検出処理(処理S10)による外部EMA検出信号と内部EMA検出信号のオアがとられて、これらのEMA検出信号の少なくとも一方でEMAフリップフロップをセットする(処理S9)。尚、内部

でEMA検出がなされた場合には、外部に対してEMAコマンドを出力する。

【0025】そして、あるタイミングを取って(処理S11)、トレースメモリ11のアドレスカウンタをタイミングET1で+1して(処理S12)、EMA発生ビットをそのカウント値で指示するアドレスのトレースメモリ11に書き込む(処理S13)。この書き込みはトレースメモリ11のバンク内のどのアドレスでEMAが発生したかを知るために行う。尚、たとえばCPUアドレスが16ビット等の時にはその上位8ビットに各種のステータス情報を設けることができる。

【0026】そして、次にEMAカウンタを+1して(処理S14)、トレースメモリ11の次のバンクに書き込みを移す。このEMAカウンタに+1をする操作は、具体的には、トレースメモリ11のアドレスカウンタの上位2ビットを用いて行う。たとえば、上位2ビットが「00」の場合はバンクB1、上位2ビットが「01」の場合はバンクB2、同様に「10」の場合はバンクB3、「11」の場合はバンクB4へ書き込みを行うというような制御を行う。尚、アドレスカウンタの上位2ビットとそれ以下のビットとは切り離されており、上位2ビットは処理S14で歩進し、下位は処理S6で歩進する。下位ビットの歩進は順次なされ、最大値となった後の最初の歩進で0より再度開始する。

【0027】このようにして、EMAの発生する毎にバンクB1からB2へ、B2からB3へと、書き込みのバンクを移して4つ目のEMAにより、トリが発生とみなし(処理S15)、トレースを停止する。

【0028】ところで上記トレースメモリ11は図3のような内容となっている。以下同図によりこのトレースメモリ11の内容について説明する。このトレースメモリ11の全容量は前記したように256KW×28bitであり、これをアドレス方向に64KWずつ4つのバンクB1～B4に分割し、ビット方向は、#00～#23の24ビットがアドレス情報エリアADE、#24～#27の4ビットがフラグエリアFREとなっている。

【0029】上記アドレス情報エリアADEには、ジャンプ元アドレス情報が書き込まれている。そして、第1のバンクB1は1回目のEMA以前のトレースの内容、第2のバンクB2は2回目のEMA以前のトレースの内容、第3のバンクB3は3回目のEMA以前のトレースの内容、第4のバンクB4は4回目のEMA以前のトレース内容を示す。

【0030】また、フラグエリアFREのうち、#24に対応するビット(25ビット目)は、有効フラグ、#25に対応するビット(26ビット目)は2Wordフラグ(略して、2Wフラグという)、#26に対応するビット(27ビット目)は内部EMAフラグ、#27に対応するビット(28ビット目)は外部EMAフラグを示している。

10

20

30

40

50

【0031】上記有効フラグは、そのフラグが“1”のとき対応するアドレス情報が有効であることを示している。また2Wフラグは、この2Wフラグに“1”が立っているとき、その直前のアドレス情報はそれを+1したアドレスが真のジャンプ元アドレスであることを示している。つまり、図3において、第2のバンクB2の2Wフラグに“1”が立っていたとすると、第1のバンクのアドレス情報(#00~#23)に+1したアドレス値が真のジャンプ元アドレスであることを示している。これは、2Wフェッチ機能(1つのアドレスアクセスで2つのアドレス情報を取り出す機能)によるものに対応させるためのものである。たとえば、CPU3からの1つのアドレス要求に対して、メインメモリ2の100番地と101番地のアドレス情報が取り出され、100番地は単なるロード命令、101番地がジャンプ命令であったとすると、プロセッサバス1上には100番地のアドレス情報しか出現しないが、実際にはプロセッサステータスにより101番地のアドレス情報も検出できる。

【0032】したがって、2Wフラグに“1”が立っていないときは、アドレス情報エリアADEのアドレス情報はそのままそれをジャンプ元アドレスとして使用し、また2Wフラグに“1”が立っているときは、その“1”の立っている1つ前のアドレス情報エリアADEに書き込まれているアドレス情報は、そのアドレス情報に+1をしたアドレス値が真のジャンプ元アドレスであることを示している。つまり、2Wフラグに“1”の立っている1つ前のアドレス情報エリアADEのアドレス値が100番地であるとする、それに+1した101番地のアドレス情報が真のジャンプ元アドレスであることを示している。

【0033】また、内部EMAフラグが“1”のときは内部EMA発生ポイントを示し、外部EMAフラグが“1”のときは外部EMA発生ポイントを示している。トレースメモリ11をこのような構成にすることにより、パソコンなど外部表示部16を外部インタフェース部14に接続して、トレース内容を画面表示して調べる際、有効フラグにより意味のない無効データを排除でき、また2Wフラグにより上記したように2Wフェッチの場合、真のジャンプ元アドレスを容易に知ることができ、また、内部・外部EMAフラグにより、そのEMAフラグの立っている直前からさかのぼってジャンプ元アドレス情報を解析することができる。

【0034】次に図5によりトレースメモリ11への具体的な書き込み例を説明する。まず、ロード命令だけの命令フェッチ(1フェッチ)の場合は、アドレス情報エリアADEに1フェッチアドレス情報が書き込まれるだけで、フラグエリアFREには何も書き込まれない(同図(a))。また、ジャンプ成立直後の命令フェッチ(J1フェッチ)の場合は、アドレス情報エリアADEには、J1フェッチアドレス情報が書き込まれ、フラグ

エリアFREにはそのアドレス情報が有効であることを示す有効フラグ(#24に対応するビット)に“1”が立つ(同図(b))。2W目実行によるJ1フェッチの場合は、アドレス情報エリアADEにJ1フェッチアドレス情報が書き込まれ、さらにフラグエリアFREの有効フラグと2Wフラグ(#25に対応するビット)にそれぞれ“1”が立つ(同図(c))。また、内部EMAの場合は、有効フラグとフラグエリアFREの内部EMAフラグ(#26に対応するビット)に“1”が立ち(同図(d))、外部EMAの場合は、有効フラグと外部EMA(#27に対応するビット)に“1”が立つ(同図(e))。

【0035】また、メモリクリアという機能は、フラグエリア(#24~#27)をクリアする機能であり(同図(f))、さらにメモリクリア(スイッチ)は、このプログラム実行履歴記憶装置の機能が正常に動作しているか否かをデバッグするもので、通常、スイッチを押すことによりクリアされるが、スイッチをデバッグモードに切り換えることにより、#24~#27にトレースメモリ11のアドレスの下位と同じ値が入っていく。たとえばトレースメモリ11のアドレス0番地には0、1番地には1というような値が入って行く。したがって、これをパソコンで読み取ることにより、機能が正常に動作しているか否かを知ることができる。

【0036】上記したようにこの実施例では、プロセッサバス上のアドレス情報をモニタし、所定のタイミングでアドレス情報を取り込むと同時に、単なる命令フェッチであるかジャンプ成立直後の命令フェッチであるかその他であるかを判定する。そして、取り込んだアドレス情報からEMA検出を行うと、トレースメモリ11のアドレスカウンタを+1してそれをトレースメモリ11に書き込んだのち、EMAカウンタを+1して、トレースメモリ11のバンクを次に移して、この移ったバンクに書き込みを行い、4回のEMA検出でトリガと判定してトレースを停止するようにしている。また、他のプロセッサからのEMA信号を検出して同様の処理を行うようにしている。

【0037】一方、上記アドレス情報の取り込みと同じタイミングでジャンプ成立直後の命令フェッチ(J1ーフ)を行う。つまり、ジャンプ命令があると、トレースメモリ11のアドレスカウンタを+1して、その+1したアドレスをトレースメモリ11に書き込むという動作を、ジャンプ命令の発生毎に繰り返す。これにより、トレースメモリ11にはジャンプ命令のあったアドレスだけを履歴として残すことができる。これに対応して、上記EMA検出時もトレースメモリ11のアドレスカウンタを+1し、そのアドレス値をそのアドレスで指示される位置に格納している。これにより、このメモリアドレス11の内容をパソコンなどに取り込み、それを画面上に表示することにより、プログラムの走行上における解

析を適確に行うことができる。

【0038】なお、上記実施例ではトレースメモリ11を4つのバンクに分割した例を示したが、これは4つに限られるものでないことは勿論であり、トレース対象装置の特性に合わせて、実装上の制約を考慮した範囲で任意に設定すれば良い。

【0039】

【発明の効果】本発明によれば、プログラムの実行状況をハードウェア的にトレースして、そのトレースした内容をトレースメモリに記憶させてプログラムの実行状況を解析する装置において、上記トレースメモリの容量をアドレス方向に複数のバンクに分割し、プログラム処理における障害発生時に、その障害を検出する毎に次のバンクにジャンプして、データ書き込みを行うよう制御するとともに、上記トレースメモリにはジャンプ時のデータを記録として残すようにしたので、障害が連続して発生した場合、複数回の連続した障害発生記録をとることができる。これにより複合的な装置障害の原因解析を行うための有効なデータを得ることができ、障害原因を短時間で究明することが可能となる。

*20

*【図面の簡単な説明】

【図1】本発明の原理を説明する構成図である。

【図2】本発明のプログラム走行軌跡記録装置の実施例を示す構成図である。

【図3】図2におけるトレースメモリの内容を示す図である。

【図4】同実施例の動作を説明するフローチャートである。

【図5】図3で示したトレースメモリへの具体的な書き込み例を示す図である。

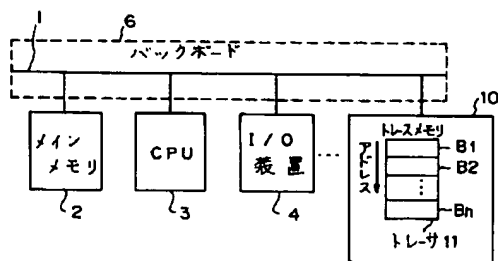
【図6】従来のトレース方式を説明するシステム構成図である。

【符号の説明】

- 1 プロセッサバス
- 2 メインメモリ
- 3 CPU
- 4 I/O装置
- 10 トレーサ
- 11 トレースメモリ

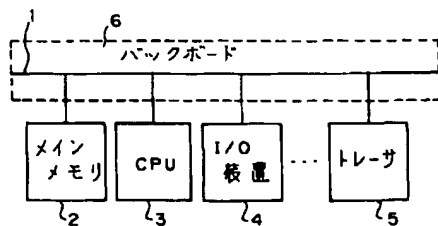
【図1】

本発明原理を説明するシステム構成図



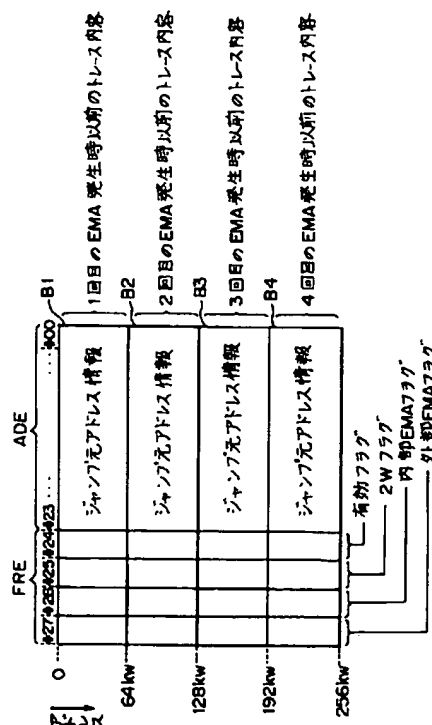
【図6】

従来のトレース方式を説明するシステム構成図



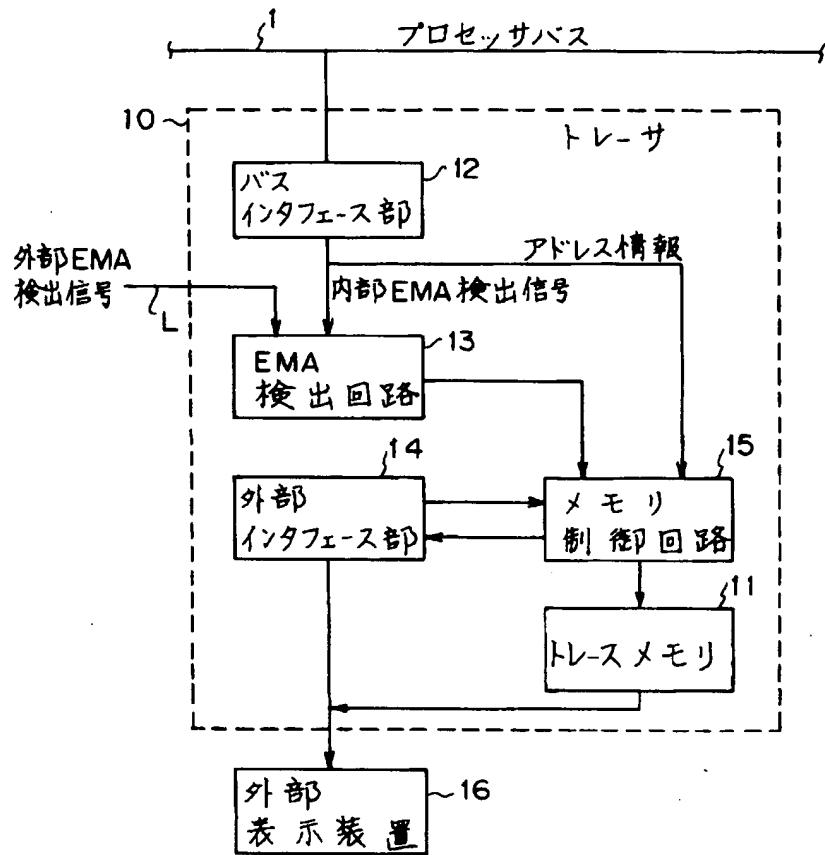
【図3】

実施例のトレースメモリの内容を示す図



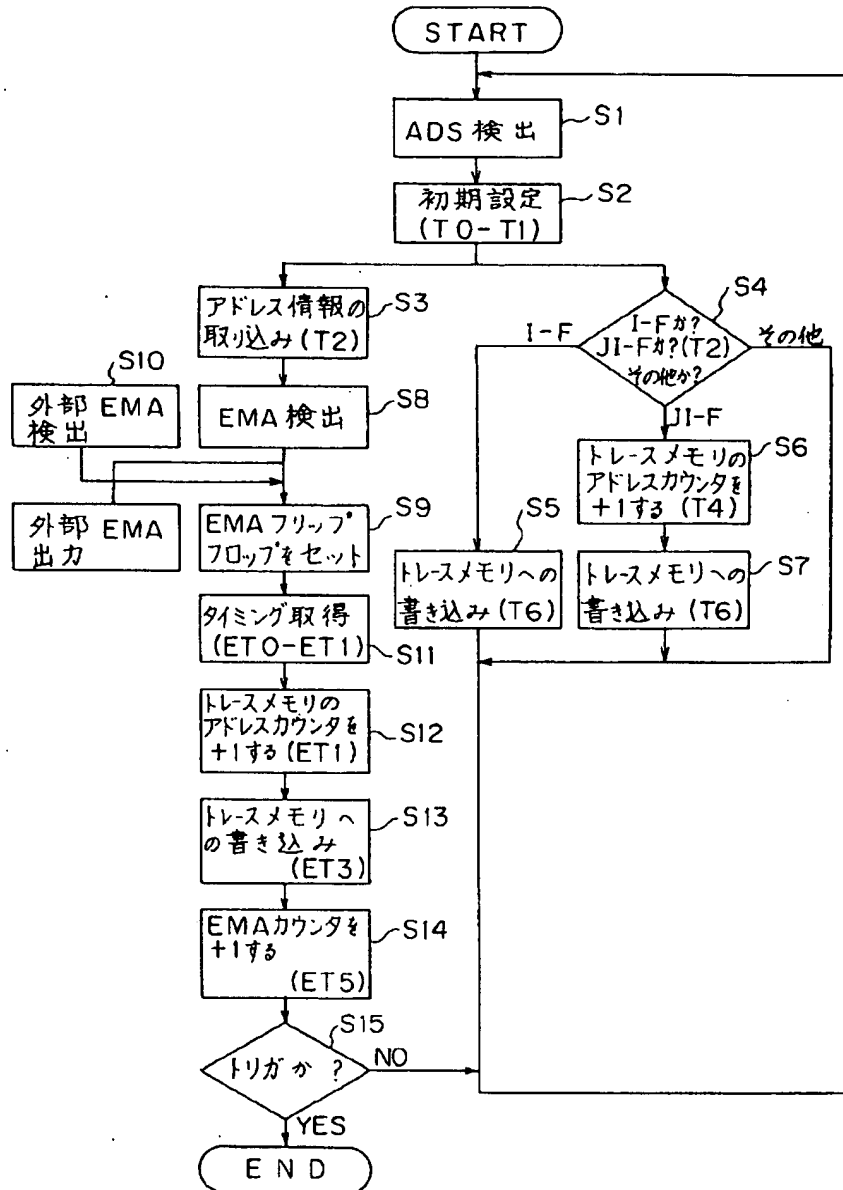
【図2】

実施例を説明する構成図



【図4】

実施例の処理動作フローチャート



【図5】

トレスメモリへの具体的な書き込み例を示す図

	FRE				ADE	
	#27	#26	#25	#24	#23, ...	#00
(a) I フェッチ					I フェッチアドレス情報	
(b) JI フェッチ	0	0	0	1	JI フェッチアドレス情報	
(c) 2W目実行による JI フェッチ	0	0	1	1	JI フェッチアドレス情報	
(d) 内部EMA	0	1	0	1		
(e) 外部EMA	1	0	0	1		
(f) メモリクリア	0					
(g) メモリクリア(スイッチ)	0				トレスメモリ11の アドレス値	

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.